

## Web-portaali pelaajayhteisölle

Antti Sopanen

Opinnäytetyö

Ammattikorkeakoulututkinto



Koulutusala Luonnontieteiden ala	
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma	
Työn tekijä(t) Antti Sopanen	
Työn nimi Web-portaali pelaajayhteisölle	
Päiväys 10.5.2011	Sivumäärä/Liitteet 30 + 10
Ohjaaja(t) Jyrki Linja	
Toimeksiantaja/Yhteistyökumppani(t) Pelaajayhteisö	
<p><b>Tiivistelmä</b></p> <p>Opinnäytetyön aiheena oli luoda dynaaminen web-portaali Starcraft 2 – tietokonepelin pelaajayhteisölle. Web-portaali mahdollistaa peliyhteisön sisäisten sarjapeliä sekä turnauksien järjestämisen sekä niiden ylläpidon. Yhteisön jäsenet voivat hallita omaa käyttäjäprofiiliaan, ladata sekä lähettää sivustolle pelitallenteita ja liittyä yhteisön omalle keskustelukanavalle.</p> <p>Tämäkaltaiselle sivustolle on tarvetta, sillä Starcraft 2:n käyttämä Battle.net – ympäristö ei mahdollista omien sarjojen tai turnauksien järjestämistä. Muokkaamalla web-portaalin pohjaa, voidaan sivustoa käyttää myös muiden harrastusten ja peliyhteisöjen organisointiin.</p> <p>Sivusto toteutettiin käyttämällä HTML-, PHP- sekä Javascript – ohjelmointikieliä sekä MySQL - tietokantaa. Sivuston dynaamisuus näkyy käyttäjän mahdollisuutena vaikuttaa suoraan sivuston sisältöön, ja olla vuorovaikutuksessa sivuston tuoman sisällön kanssa. Sivuston näyttämä tieto ei ole kiinteästi kirjoitettu palvelimella oleviin PHP-tiedostoihin, vaan suuri osa sivuston näyttämästä tiedosta haetaan palvelimella toimivasta MySQL-tietokannasta.</p>	
Avainsanat WWW, PHP, MySQL, kirjautumisjärjestelmä	

Field of Study Natural sciences			
Degree Programme Degree Programme in Computer Science			
Author(s) Antti Sopenen			
Title of Thesis Web portal for a player community			
Date	10.5.2011	Pages/Appendices	30 + 10
Supervisor(s) Jyrki Linja			
Project/Partners Player community			
<p><b>Abstract</b></p> <p>The goal of the thesis was to create a dynamic web portal for a Starcraft 2 - PC game player community. The web portal would make it possible to arrange and maintain in-house league games and tournaments. Community members would be able to manage their user profiles, download and upload recorded gaming sessions on the website and join to the community's own chat channel.</p> <p>Sites like this are in demand because the Starcraft 2 Battle.net service does not provide support for organizing in-house leagues or tournaments. By modifying the basis of the web portal, the site can also be used to support other hobbies and player communities.</p> <p>The website was created by using HTML, PHP and JavaScript programming languages and MySQL database. The dynamic structure of the website provides an opportunity for a user to directly have an effect on the site's content and to interact with it. The content of the web pages is not hard coded into PHP files. A large part of the page content is retrieved from the MySQL database server.</p>			
<p><b>Keywords</b></p> <p>WWW, PHP, MySQL, login system</p>			

## SISÄLTÖ

1	JOHDANTO.....	6
2	STARCRAFT 2 – WINGS OF LIBERTY .....	7
2.1	Battle.net .....	7
3	SIVUSTON SUUNNITTELU .....	8
3.1	Sivuston käyttötarkoitus.....	10
3.2	Sisällön määrittely .....	11
4	SIVUSTON TOTEUTUS.....	12
4.1	Käytetyt ohjelmat sekä ohjelmointikielet.....	13
4.1.1	HTML ja CSS .....	13
4.1.2	PHP.....	13
4.1.3	JavaScript .....	14
4.1.4	SQL.....	14
4.2	MySQL-tietokanta.....	14
4.3	Sivuston rakenne.....	16
4.4	Kirjautumisjärjestelmä.....	17
4.5	Battle.net - tilastojen haku.....	19
4.6	Sarjataulukko.....	20
4.7	Replay .....	24
4.8	Chat.....	26
5	SIVUSTON TESTAUS.....	27
6	POHDINTA.....	28
	LÄHTEET.....	29

## LIITTEET

- Liite 1 kirjautuminen.php
- Liite 2 Funktion ajastus.
- Liite 3 Sarjataulukoiden järjestyksen käsittelykoodi.
- Liite 4 Sarjaotteluiden ylläpitotoimet.
- Liite 5. Pelitallenteiden tiedonhaku ja tulostus.

## 1 JOHDANTO

Opinnäytetyön aiheena on luoda dynaaminen web-portaali Starcraft 2 – tietokonepeelin pelaajayhteisölle. Web-portaalin avulla yhteisön jäsenet voivat esitellä itsensä sivuston käyttäjille, pelata yhteisön jäsenien välisiä sarjapelejä, sekä järjestää omia turnauksia. Lisäksi sivusto mahdollistaa pelitallenteiden jakamisen. Yhteydenpitämistä varten sivustolta pääsee liittymään yhteisön omalle keskustelukanavalle.

Web-portaali on luotu käyttämällä PHP-, HTML-, CSS- sekä JavaScript – ohjelmointikieliä. Sivuston luominen vaati minulta itsenäistä opiskelua, sillä en ole opiskellut PHP-ohjelmointikieltä aikaisemmin. Sivusto käyttää MySQL - tietokantaa, jota hallitaan MySQL - hallintajärjestelmällä.

Tämänkaltaiselle sivustolle on tarvetta, sillä Starcraft 2:n käyttämä Battle.net – ympäristö ei mahdollista omien sarjojen tai turnauksien järjestämistä. Muokkaamalla web-portaalin pohjaa, voidaan sivustoa käyttää myös muiden harrastusten ja peliyhteisöjen organisointiin.

## 2 STARCRAFT 2 – WINGS OF LIBERTY

Starcraft 2 – Wings of Liberty on Blizzard Entertainment -nimisen peliyhtiön tekemä ja julkaisema reaaliaikainen tietokonestrategiapeli. Starcraft 2- Wings of Liberty julkaistiin vuonna 2010. Starcraft 2 on jatko-osa huippusuositulle Starcraft – tietokonepelille joka julkaistiin vuonna 1998. Starcraft 2 sisältää haastavan yksinpelin hyvin kirjoitetulla tarinalla, sekä nopeatempoisen verkkomoninpelin. Pelaajalla on valittavana kolme erilaista ”rotua”. Jokaisella rodulla on omat heikkoudet ja vahvuudet, ja niiden tunteminen voi olla ratkaiseva tekijä voiton ja häviön välillä. (Blizzard Entertainment, 2011) Lisätietoa Blizzard Entertainmentista ja heidän peleistään löytyy Blizzardin kotisivuilta <http://eu.blizzard.com>.

Starcraft 2:ssa pelaajat keräävät työläisillä mineraaleja, joita käytetään oman tukikohdan sekä armeijayksikköjen tekemiseen ja kehittämiseen. Peli vaatii pelaajalta strategista ajattelua ja nopeaa ratkaisuntekokykyä. Moninpeliä pelatessa pelataan toisia ihmisiä vastaan. Peli päättyy, kun toinen pelaajista luovuttaa, tai kaikki pelaajan rakennukset ovat tuhoutuneet.

### 2.1 Battle.net

Battle.net on Blizzard Entertainmentin luoma verkkopelipalvelu. Sen tavoitteena on yhdistää kaikki Blizzard – pelaajat yhteen tehokkaaseen ja kehittyneeseen verkkopelipalveluun.

Battle.net yhdistää miljoonia verkkopelaajia ympäri maailman. Se tarjoaa pelaajille käyttäjäystävällisen sekä vakaan alustan verkkopelaamiselle. Battle.net sisältää älykkään ja tarkan vastustajahaun, joka takaa pelaajille hauskan ja reilun verkkopelikokemuksen. (Blizzard Entertainment, 2011)

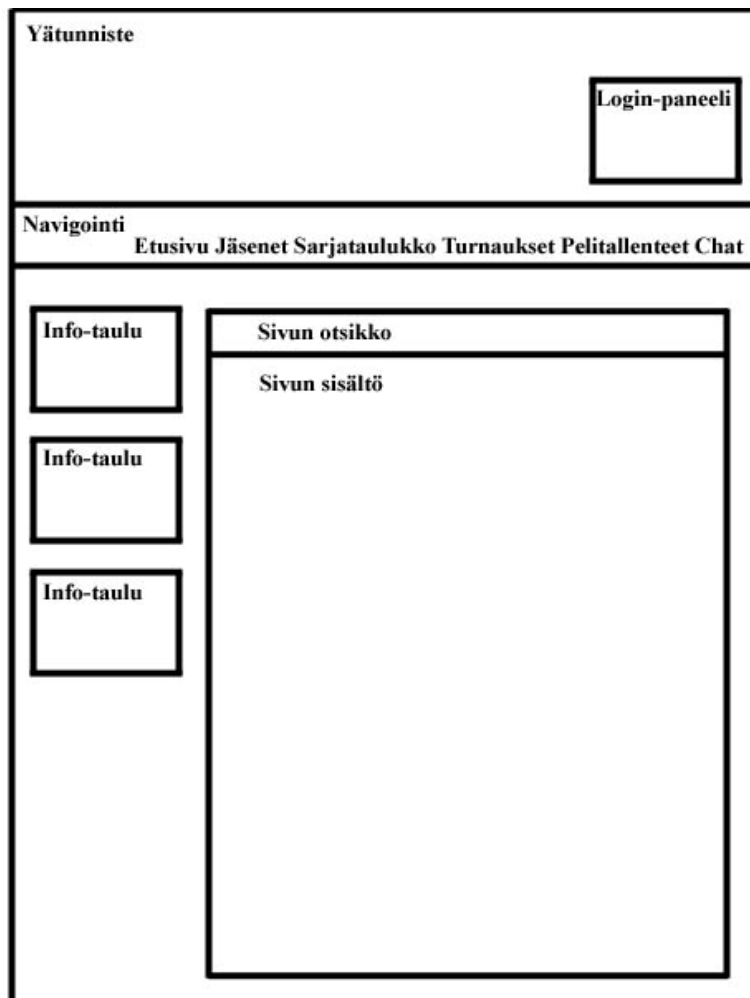
### 3 SIVUSTON SUUNNITTELU

Sivuston toimeksiantajana toimi pieni pelaajayhteisö, joka toivoi saavansa omalle Starcraft 2 - yhteisölleen dynaamisen web-portaalin, jonka avulla he voivat järjestää jäsenien kesken sarjaotteluita sekä turnauksia.

Sivustoa suunnitellessa otettiin huomioon niin pelaajayhteisön toiveet sekä tarpeet, kuin omat tavoitteeni ja toiveeni opinnäytetyön tekemisen suhteen. Opinnäytetyön tavoitteena on kartuttaa ammattitaitoani ja lisätä web-ohjelmoinnin tietotaitoani. Minulla ei ole aikaisempaa kokemusta PHP-ohjelmoinnista, joten halusin käyttää opinnäytetyön tuomat mahdollisuudet sen itsenäiseen opiskeluun. Lähtökohtana sivuston tekemiselle oli, että sivustolla käytettäisiin mahdollisimman vähän valmiita lisäosia sekä palveluita kuten sisällönhallintajärjestelmiä. Tämä lähestymistapa palveli oman tietotaitoni karttumista, sekä loi haasteita sivuston tekemiseen.

Sivuston rakennesuunnitelmassa (Kuva 1) on huomioitu sivuston käyttötarkoitus. Sivuston ylätunnisteeseen on sijoitettu kirjautumispaneeli, jonka avulla käyttäjä voi käyttää sivuston kirjautumisjärjestelmää. Sivustolle kirjautuminen sallii käyttäjälle pääsyn sivuille, joille pääsevät vain jäsenet. Navigointipalkki sisältää sivuston kuusi pääsivua, joiden kautta vierailija voi tutustua sivuihin. Näiltä pääsivuilta löytyvät yhteisön jäsenille tarkoitetut linkit sivuston ylläpitosivuille.

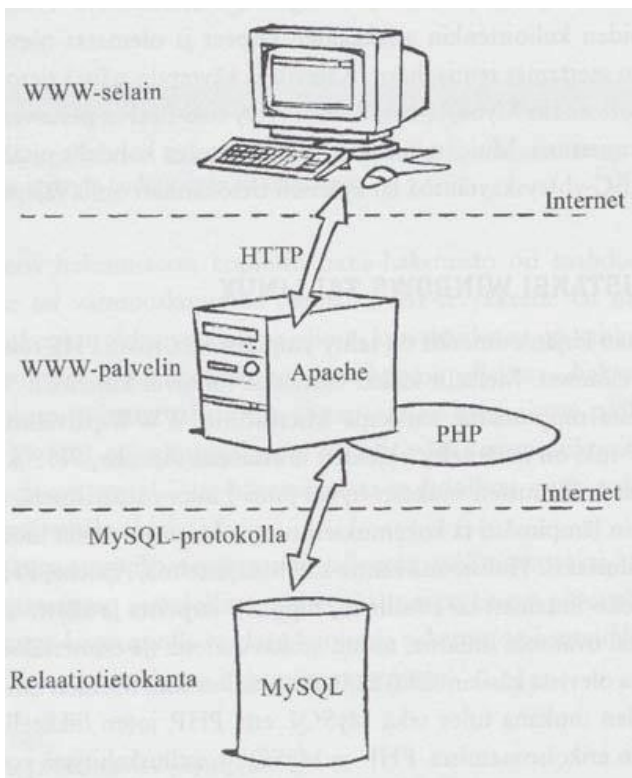




Kuva 1. Sivuston rakennesuunnitelma.

Rakennesuunnitelmassa näkyvät info-taulut näyttävät koosteen sivuston tuottamista tiedoista. Kukaan info-taulu sisältää eri tietoja, kuten sarjataulukon, viimeisimmät pelatut ottelut sekä Battle.net -tilastot. Nämä info-taulut löytyvät jokaiselta sivuston sivulta. Info-taulujen oikealle puolelle sijoittuu sisältö-kenttä, joka sisältää valitun sivun esiin tuomat tiedot.

Sivuston sisältö tuotetaan dynaamisesti, jolloin sivuston näyttämä tieto ei ole kiinteästi kirjoitettu palvelimella oleviin PHP-tiedostoihin, vaan suuri osa sivuston näyttämästä tiedosta haetaan palvelimella toimivasta MySQL-tietokannasta. Portaalin luomiseen käytetyt PHP-ohjelmointikieli sekä MySQL-tietokanta mahdollistavat sivuston dynaamisen sisällön, mutta asettaa samalla vaatimuksia kotisivutilan palveluntarjoajalle. Palveluntarjoajan kotisivupalvelun on tuettava PHP-ohjelmointikieltä, sekä tarjottava MySQL-tietokantaympäristö (Kuva 2). Sivuston dynaamisuus näkyy käyttäjän mahdollisuutena vaikuttaa suoraan sivuston sisältöön, ja olla vuorovaikutuksessa sivuston tuoman sisällön kanssa.



Kuva 2. Järjestelmien käyttäjien, WWW-palvelimen, PHP:n ja MySQL - tietokannan suhde toisiinsa. (Heinisuo, 2004)

Rakennesuunnitelmassa on pyritty ottamaan huomioon rakenteen joustavuus. Tarkoituksena on luoda sellainen sivuston rakenne, sekä taustalla toimivat toiminnot, että niitä voitaisiin käyttää jatkossa myös muilla harraste- tai urheilulajisivuilla.

### 3.1 Sivuston käyttötarkoitus

Sivustoa käyttävät Starcraft 2 –tietokonepelin pelaajayhteisö. Sivuston avulla pelaajayhteisön jäsenet voivat kommunikoida keskenään, järjestää yhteisön sisäisiä turnauksia sekä pelata sarjaotteluita. Tämänkaltaiselle sivustolle on tarvetta, sillä Starcraft 2:n käyttämä Battle.net – ympäristö ei mahdollista omien sarjojen tai turnauksien järjestämistä.

Starcraft 2 tallentaa käydyistä otteluista pelitallenteen, joita yhteisön jäsenet voivat sivuston avulla jakaa keskenään. Näiden pelitallenteiden avulla yhteisön jäsenet voivat katsoa käydyn ottelun tapahtumat jälkikäteen. Näin pelaajat voivat jakaa mielestään hyvät, tai mielenkiintoiset pelit muille katsottavaksi.

Sivuston avulla pelaajayhteisö saa näkyvyyttä ja tätä kautta voi laajentua halutesaan. Tiedettävästi vastaavanlaisia sivustoja ei ole tarjolla yksittäisille pelaajayhteisöille.

### 3.2 Sisällön määrittely

Sivuston toivotuin ominaisuus oli mahdollisuus pelata omaa suljettua sarjaa, ja tätä kautta kilpailla oman peliyhteisön jäsenien kesken paremmuudesta. Lisäksi toivottiin, että käyttäjät näkisivät yhteisön jäsenien pelinsisäiset sarjataulukkotiedot, kuten suosituimman rodun, liigan, pisteet, voitot, häviöt sekä voittoprosentin. Toiminto helpottaa pelin oman Battle.net - tilastojen vertailua, sillä jokainen pelaaja kilpailee hyvin todennäköisesti eri sarjataulukossa pelin huiman pelaajamäärän takia. Sivustolla näkyvien Battle.net – tilastojen avulla yhteisön jäsenet voivat kilpailla myös Battle.net - sarjataulukon sijoituksista.

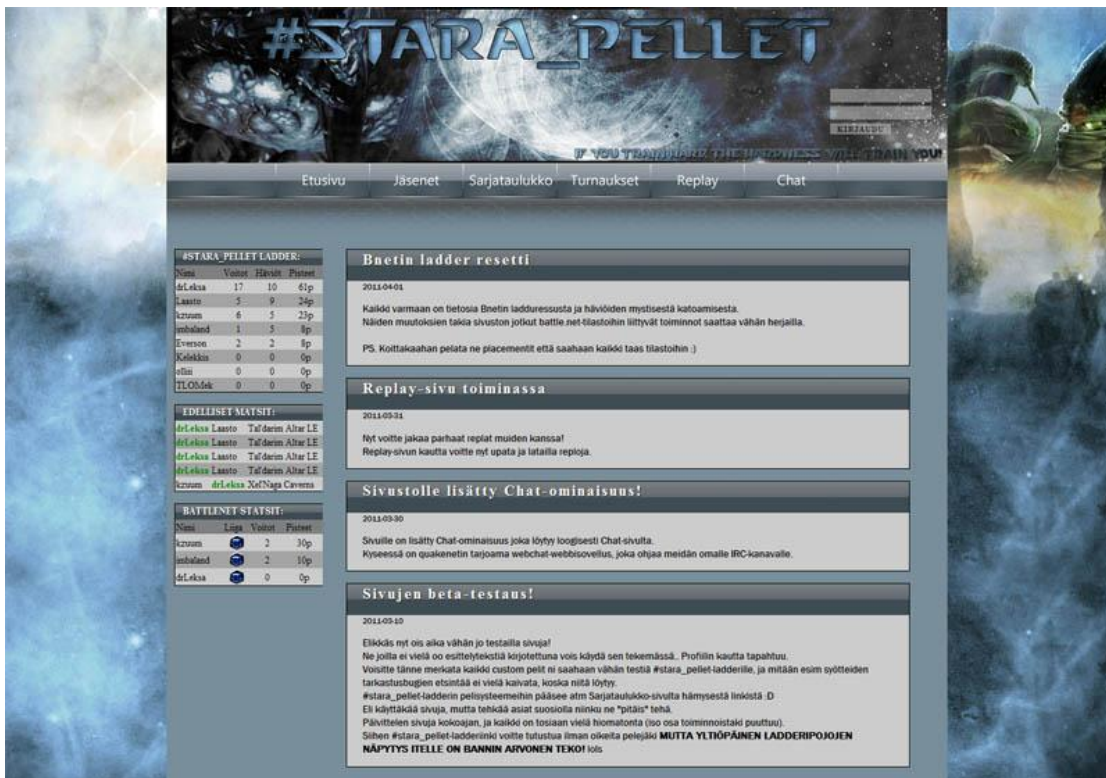
Tietoturvallisuuden takaamiseksi sivustolla vieraileva henkilö ei voi rekisteröityä suoraan yhteisön jäseneksi, sillä sivuston ylläpitotoiminnot olisivat ilkeiden armoilla. Kun vierailija kiinnostuu yhteisön toiminnasta, hän voi liittyä yhteisön omalle keskustelukanavalle sivustolta löytyvän Chat-sivun kautta ja ilmaista kiinnostuksensa yhteisöä kohtaan.

Sivuston käytettävyyteen on panostettu siten, että sarjaotteluiden luominen sekä tulosten kirjaaminen olisi käyttäjälle mahdollisimman vaivatonta. Sarjaotteluiden ylläpidon sujuvuus on ensisijassa sivuston toimivuuden kannalta, sillä pelaajat pelaavat usein useita otteluita peräkkäin. Käyttäjät haluavat käydä pelattujen otteluiden välissä nopeasti sivustolla kirjaamassa tuloksen, sekä kirjaamassa sivustolle seuraavan ottelun tulosta odottamaan.

## 4 SIVUSTON TOTEUTUS

Ennen varsinaisen toteutuksen aloitusta sivustolle täytyi hankkia web-hotelli, joka tarjoaisi kotisivutilan, PHP5-tuen, sekä MySQL-tietokantapalvelun. Palveluntarjoajia vertaillen valikoitui arkku.net – palveluntarjoaja, joka myöntää kotisivupalvelun edellä mainituilla tuilla, ja vieläpä ilmaiseksi. Palvelun hakijalta vaadittiin palvelun saamiseksi hakulomakkeen täyttö, jonka he ystävällisesti hyväksyivät jo samana päivänä kuin hakulomake oli lähetetty. Sivustolle ei hankittu maksullista domainia, joten sivusto löytyy toistaiseksi osoitteesta <http://kzoom.arkku.net/starapellet/>. Osoite muuttuu mahdollisesti tulevaisuudessa.

Kotisivutilan löytymisen jälkeen sivuston toteutus alkoi ripeästi. Sivustolle toteutettiin suunnitelman mukainen ulkoasu käyttämällä pääosin HTML:n div-elementtejä, joiden sisään upotettiin luodut graafiset elementit (Kuva 3). Sivuston ulkoasutoteutuksen todellista toimivuutta päästiin testaamaan vasta, kun sivustolle oli luotu ensimmäiset tietokantataulut sekä PHP-sivu, joka hakee tietokannassa olevat tiedot sivulle näytettäväksi. Tämä siksi, että kyseessä on dynaamisesti toimiva sivusto, jonka sisältö ja tätä kautta ulkoasu muuttuu tietokannan tietomäärän muuttuessa.



Kuva 3. Sivuston ulkoasu.

#### 4.1 Käytetyt ohjelmat sekä ohjelmointikielet

Toteutuksessa käytettiin mahdollisimman paljon avoimen lähdekoodin ohjelmia. Avoimeen lähdekoodiin perustuvat ohjelmat on julkaistu avoimen lähdekoodin lisenssillä. Lisenssin käyttöehdoissa sallitaan ohjelman vapaa jaettavuus. Ohjelman lisäksi jaetaan myös lähdekoodi, joka mahdollistaa ohjelman vapaan jatkokehityksen. (Open Source Initiative, 2011).

Sivuston web-ohjelmointiin käytettiin GPL-lisenssiin (General Public License) pohjautuvaa Notepad++ teksti- ja lähdekoodieditoria. Notepad++ tukee useita eri ohjelmointikieliä, mukaan lukien opinnäytetyöhön tarvittavat HTML-, CSS-, PHP- sekä JavaScript -ohjelmointikielet. Näin ollen Notepad++ oli luonteva valinta koodieditoriksi.

Tiedostojen siirto palvelimen sekä oman tietokoneen välillä tapahtui GPL-lisensoidun FileZilla FTP-asiakasohjelman avulla. Sivustoa luodessa käytössä ei ollut varsinaista testiympäristöä, joten testaus täytyi suorittaa sivustolle tilatulla palvelimella. Testiympäristön puutteen vuoksi palvelimen ja kehitysympäristön välillä tapahtui koko sivuston kehitystyön ajan tiedostojen siirtoa, joka osaltaan hidasti sivuston rakentamista.

Sivuston graafinen ilme on luotu Adobe Photoshop Elements 7.0 – kuvankäsittelyohjelmalla. Adoben kehittämä Photoshop – tuoteperhe on suunnattu digitaaligrafiikan tuottamisen ammattilaisille. (Adobe Systems Incorporated, 2011)

##### 4.1.1 HTML ja CSS

HTML (Hypertext Markup Language) on olennainen osa www-sivujen rakentamista. HTML on merkkauskieli, jolla kuvataan sivuston teksti sekä rakenne. Sivuston tyyliasettelut voidaan tehdä HTML:llä, mutta sitä helpottamaan sivustolla on käytetty CSS (Cascading Style Sheets) – merkintäjärjestelmää. CSS on kieli, jolla voidaan määritellä muun muassa www-sivujen ulkoasu. Sivuston ulkoasua voidaan muuttaa helposti jälkikäteenkin muokkaamalla sivuston käyttämää CSS – tyylitiedostoa. (Korpela, 2003)

##### 4.1.2 PHP

PHP (Hypertext Preprocessor) on yksi kätevimmistä sekä nopeimmista tekniikoista dynaamisten verkkopalvelujen tekemiseen. PHP on alun perin tarkoitettu WWW-pohjaisten sovellusten tekoa helpottavana komentokokoelmana, mutta toimii nykyään

jo täysiverisenä ohjelmointikielenä. PHP:n kehitys on ollut erittäin nopeaa, ja PHP versio viiden tuki on jo yleinen näky WWW-palvelimilla. (Heinisuo, 2004)

PHP-ohjelmointikielen komentoja voidaan upottaa suoraan HTML-sivujen sisään. PHP on tulkittava kieli, eli WWW-sivun sisällä oleva PHP-koodi ajetaan jokaisella kerralla, kun WWW-palvelin lähettää sivun selaimelle. PHP-koodi ajetaan aina palvelimella ennen kuin sivu lähetetään selaimeseen. Tästä syystä sivujen käyttäjät eivät näe sivun sisältämää PHP-koodia, kun he katsovat sivun HTML-koodia selaimessaan. (Heinisuo, 2004)

#### 4.1.3 JavaScript

JavaScript on yleiskäyttöinen ohjelmointikieli, jolla voidaan luoda selainskriptejä (client-side scripting). JavaScriptillä voidaan tehdä web-sivuja, jotka reagoivat käyttäjän sivulla tekemiin toimenpiteisiin, kuten hiiren napsautuksiin, syöttötietojen kirjoittamiseen ja sivulla liikkumiseen. (Korpela, Linjama 2005)

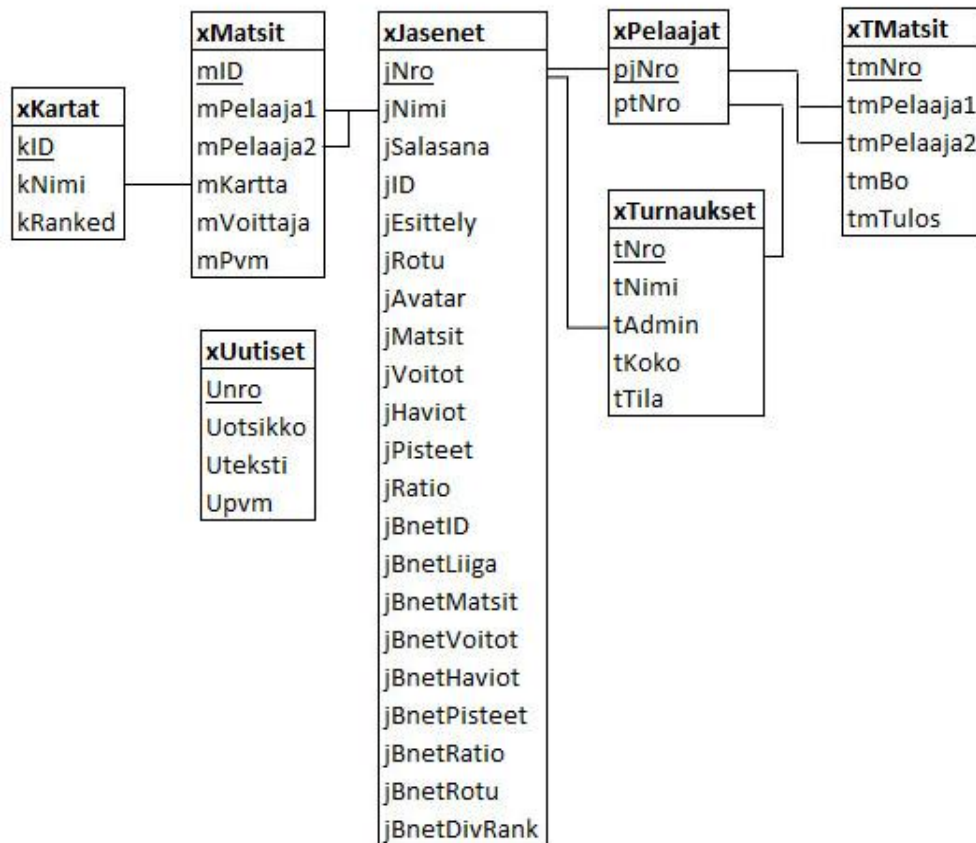
JavaScript on yksi monista selainskriptien tekemiseen tarkoitettu käytetyistä kielistä. JavaScript on kuitenkin ylivoimaisesti laajimmin selainten tukema ja sen takia yleisimmin käytetty ohjelmointikieli selainskriptien luomiseen. (Korpela, Linjama 2005)

#### 4.1.4 SQL

SQL (Structured Query Language) on monipuolinen ja voimakas relaatiotietokantakieli. SQL:llä voidaan lukea ja päivittää tietokantaa sekä poistaa ja lisätä rivejä. SQL-koodi voidaan upottaa ohjelmointikieleen, tai käyttää vuorovaikutteisesti päätteeltä tai työasemalta. (Hovi, Huotari, Lahdenmäki 2003)

#### 4.2 MySQL-tietokanta

Tilattu Arkku.net web-hotelli sisältää phpMyAdmin – hallintapaneelin. PhpMyAdmin on PHP:llä kirjoitettu ilmainen ohjelma, joka mahdollistaa MySQL-tietokantojen hallinnoinnin selaimen välityksellä. (phpMyAdmin devel team, 2011)



Kuva 4. Tietokannan relaatiokaavio.

Sivuston käyttämä tietokanta on rakennettu käyttämällä phpMyAdmin - hallintapaneelia. Tietokanta rakentuu jäsentaulun ympärille (Kuva 4). Puhuttaessa jäsentaulusta, tarkoitetaan kuvassa esitettyä xJasenet – taulua. Jäsentauluun tallennetaan kaikki sivuston tarvitsemat tieto ja statistiikka jäsenestä. Taulun kentät jNimi sekä jSalasana toimivat käyttäjän käyttäjätunnuksena sekä salasanan sivustolle kirjaututtaessa. Jäsentaulun sisältämät jEsittely, jRotu sekä jAvatar ovat käyttäjän profiilitietoja, jotka näkyvät esimerkiksi jäsenesittelyssä. Muut taulun tiedot liittyvät sivustolla näkyviin sarjataulukoihin joihin haetaan pelaajatiedot tietokannasta.

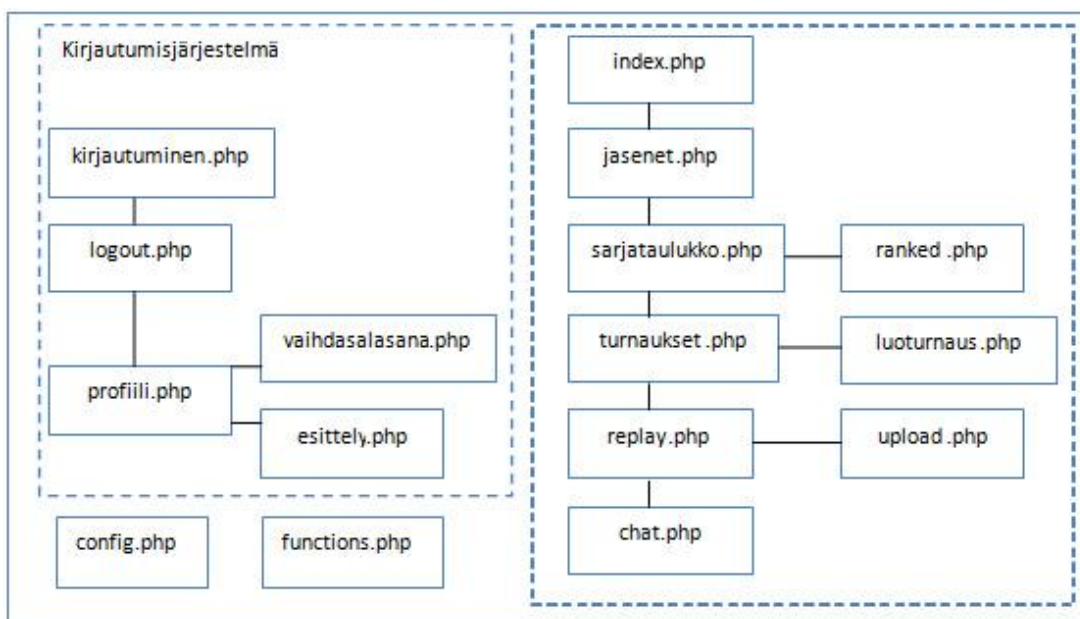
xMatsit – tauluun tallennetaan jäsenien käymät sarjaottelut. Sarjaottelun kirjaaja valitsee sivulla kaksi pelaajaa, sekä pelattavan kartan. Pelaajalistaan listataan xJasenet – taulussa olevat yhteisön jäsenet. Kartat haetaan sivulla toisesta tietokannan taulusta nimeltä xKartat, johon on tallennettu kartan nimi, sekä tieto siitä, onko kartta pelin virallinen kartta. xMatsit – tauluun tallennetaan käyttäjän valitsevat tiedot, sekä kirjauksen päivämäärä. Sarjaottelun luomisen, sekä pelatun pelin jälkeen käyttäjä käy päivittämässä ottelun tietoihin voittajan.

Turnausjärjestelmä on vasta kehitteillä, ja tietokantaratkaisu ei välttämättä ole tällä hetkellä täysin toimiva. Relaatiokaavion (Kuva 4) esittämässä ratkaisussa xTurnaukset – tauluun tallennetaan turnaukseen liittyvät tiedot, kuten turnauksen nimi, ylläpitäjä, pelaajamäärä sekä turnauksen tila. Turnauksen tilaa muuttamalla ylläpitäjä voi merkitä turnauksen alkaneeksi, jolloin turnaukseen ei voida enää ilmoittautua. Turnaukseen ilmoittautuneet pelaajat kerätään xPelaajat – tauluun, johon tallennetaan turnauksen numero sekä jäsennumero. xTMatsit – taulu kerää turnaukseen ilmoittautuneista pelaajista ottelulle pelaajaparit, jotka näkyvät sivustolla turnauksen turnauspuussa. xTMatsit – taulussa olevaan tmBo – kenttään merkitään turnauskierroksen ”paras x:stä” ottelumäärä. Turnausottelun päätteeksi voittanut pelaaja käy merkkauksessa tuloksen, jonka hävinnyt pelaaja vahvistaa. Tuloksen merkkauksen jälkeen hävinnyt pelaaja poistetaan turnauksen pelaajista, ja uuden turnauskierroksen alkaessa haetaan uudet otteluparit xTMatsit – tauluun.

#### 4.3 Sivuston rakenne

Sivuston rakenne jakautuu kahteen osaan (Kuva 5). Niin sanottuun kirjautumisjärjestelmään, joka sisältää kirjautumisjärjestelmälle yleiset toiminnot, kuten kirjautumisen, uloskirjautumisen sekä kirjautuneen henkilön profiilihallinnan.

Jokainen portaalin sivu käyttää kirjautumisjärjestelmää. Sivuston käyttäjälle näkymättömät sivut config.php sekä functions.php sisältävät sivustolle luotuja toimintoja sekä tietokannan yhteystiedot.



Kuva 5. Sivuston rakenne.



#### 4.4 Kirjautumisjärjestelmä

Sivuston kirjautumisjärjestelmä perustuu PHP:n mahdollistamaan istunto-tekniikkaan. Istunto on tapa tallentaa tietoja, joita voidaan käyttää sivuston eri sivuilla. Toisin kuin evästeitä käytettäessä, tietoa ei ole tallennettu käyttäjän tietokoneelle, vaan tieto säilytetään palvelimella. Istuntoon tallennetut muuttujat eroavat normaaleista muuttujista siinä mielessä, että muuttujia ei kuljeteta erikseen jokaiselle uudelle sivulle, vaan ne haetaan istunnon avauksen yhteydessä jokaiselle sivulle. (Bradley, 2011)

Sivusto seuraa jatkuvasti käyttäjän kirjautumista ja muokkaa selattavia sivuja sen mukaan, onko käyttäjä kirjautuneena sivustolle vai ei. Kirjautumisen seurantakoodi (Kuva 6) sijaitsee jokaisen sivun banner – divin sisällä.

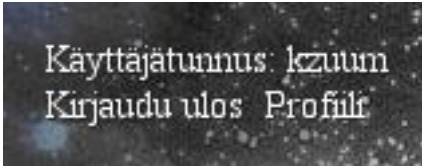
```
<div id="banner">
  <?php
    if(!isset($_SESSION['SESS_ID']) || (trim($_SESSION['SESS_ID']) == '')){
      print("<div id='login'>
        <form id='loginForm' name='loginForm' method='post' action='kirjautuminen.php'>
          <table width='150' border='0' align='center' cellpadding='0' cellspacing='0'>
            <tr>
              <td width='150'><input name='kayttajatunnus' type='text' class='textfield' id='kayttajatunnus' /></td>
            </tr>
            <tr>
              <td><input name='salasana' type='password' class='textfield' id='salasana' /></td>
            </tr>
            <tr>
              <td><input id='kirjautu' type='submit' name='Submit' value='KIRJAUDU' /></td>
            </tr>
          </table>
        </form>
      </div>");}
    else {
      print("<div id='login'>
        <font id='flogin'> Käyttäjätunnus: " . $_SESSION['SESS_NIMI'] . "</font>";
      print("<br/>
        <font id='flogin'><a href=logout.php target='_self'>Kirjautu ulos</a>&nbsp;
        <a href=profiili.php target='_self'>Profiili</a></font>
      </div>");}
    ?>
  </div>
```

Kuva 6. Kirjautumisen seuranta.

Jos sivuston käyttäjä ei ole kirjautuneena sisään, sivuston ylätunnisteessa näytetään kirjautumislomake (Kuva 7), johon tunnukset omaava käyttäjä täyttää käyttäjätunnuksensa sekä salasansa. Jos käyttäjä on kirjautuneena sisään, ylätunnisteen kirjautumislomakkeen tilalla on painikkeet käyttäjäprofiilin muokkaukseen sekä uloskirjautumiseen (Kuva 8).



Kuva 7. Kirjautumislomake.



Kuva 8. Näkymä kirjautuneena.

Kirjautuminen.php sisältää käyttäjän syötteiden tarkastukset, ja vertaa näitä tietokannassa oleviin jäsentietoihin. Jos kirjautuja on syöttänyt oikean käyttäjänimen ja salasanan, hänelle luodaan sivustolle istunto, jonka tietoihin tallennetaan käyttäjän jäsennumero, käyttäjänimi sekä salasana. Koko kirjautuminen.php – sivun koodi löytyy raportin lopussa liitteenä (LIITE 1).

```
//Kirjautuminen onnistunut
session_regenerate_id();
$jasen = mysql_fetch_assoc($result);
$_SESSION['SESS_ID'] = $jasen['jNro'];
$_SESSION['SESS_NIMI'] = $jasen['jNimi'];
$_SESSION['SESS_SALASANA'] = $jasen['jSalasana'];
header("location: ".$_SERVER['HTTP_REFERER']);
// palataan sivulle, jolta kirjautumiskutsu tuli
exit();
```

Osa portaalin sivuista on tarkoitettu ainoastaan yhteisön jäsenille, joten näille sivuille pääsevät vain kirjautuneet käyttäjät. Yhteisön jäsenille rajoitetut sivut sisältävät pääosin sivustolle elintärkeitä ylläpidollisia toimintoja, kuten sarjapelien sekä turnauksien ylläpidon. Yhteisön jäsenille rajoitettujen sivujen lähdekoodiin on kirjoitettu pyyntö tarkastus.php – nimiselle PHP - koodisivulle.

```
<?php
require_once('tarkastus.php');
?>
```

Tarkastus.php - sivu sisältää istuntotarkastuksen, jonka ajon jälkeen käyttäjä ohjataan joko virheilmoitussivulle, tai hänet päästetään valitsemalleen sivulle.

```
<?php session_start();
if(!isset($_SESSION['SESS_ID']) || (trim($_SESSION['SESS_ID'])
== '')) {
    header('location: paasy_kielletty.php');
    exit();
}
?>
```

Kun käyttäjä on kirjautuneena sisään, hänellä on mahdollisuus kirjautua sivustolta ulos (Kuva 8). Uloskirjautuminen tapahtuu koodissa istunnon tuhoamisella, joka poistaa istuntotiedoston palvelimelta.

```
<?php session_start();
//Poistetaan tallessa olevat tiedot ja palataan edelliselle
sivulle
unset($_SESSION['SESS_ID']);
unset($_SESSION['SESS_NIMI']);
session_destroy();
header("location: " . $_SERVER['HTTP_REFERER']);
exit();
?>
```

#### 4.5 Battle.net - tilastojen haku

Battle.net ei tarjoa fanisivustoille palvelua jonka kautta fanisivustot saisivat kätevästi pelaajien tilastot käyttöönsä. Palvelu on kuitenkin saatavilla SC2Ranks.com – sivuston tarjoamana (SC2 Ranks, 2011). Jäsenien Battle.net – tilastot haetaan SC2Ranks – sivuston tarjoaman ohjelmointirajapinnan (engl. Application programming interface, API) avulla. SC2Ranks.com – sivustolta saatava PHP:llä kirjoitettu API sisältää valmiit funktiot tietojen hakemiseen SC2Ranks.com – sivustolta. API tukee montaa erilaista hakumenetelmää, mutta lähiaikoina tapahtuneiden Battle.net muutosten takia hakukriteereitä on jouduttu hieman muuttamaan. Jotta pelaajatiedot saadaan haettua, on tietokantaan pitänyt tallentaa yhteisön jäsenien niin sanotut Battle.net – numerot sekä Battle.net - käyttäjätunnukset. Battle.net – käyttäjätunnukset toimivat samalla jäsenien käyttäjätunnuksena yhteisön sivustolla.

SC2Ranks API:n sisältämät PHP – tiedostot on tallennettuna sivuston palvelimelle. Tilastojen hakuun on luotu oma PHP – tiedosto, joka kääntää API:n PHP – tiedostoa hakemaan pelaajatilastot. Hakuprosessin aikana pelaajatilastot tallennetaan kunkin pelaajan kohdalla MySQL – tietokantaan.

```
<?php
include ('HaeSC2Ranks.php');
$statsit = NULL;
//päivitetään SC2Ranksista haetut tilastot
$statsit = cache_function('HaeSC2Ranks');
//Cache_function ajastaa funktion. Tilastot haetaan enintään
tunnin välein.
```

Koska pelaajatiedot käydään hakemassa toiselta sivustolta, nousevat oman sivuston latausajat huomattavasti. Tästä syystä ei olisi mielekästä hakea tilastoja joka kerta kun sivustolla ladataan sivuja. Ongelma on ratkaistu ajastamalla (caching) Battle.net -tilastojen hakufunktio. Hakufunktio on ajastettu päivittämään pelaajatilastot enintään tunnin välein. Cache\_function() - funktiokutsu tallentaa hakufunktion saamat tiedot palvelimelle väliaikaistiedostoksi, joka päivitetään ajastuksen mukaan. Ajastusfunktio tarkkailee väliaikaistiedoston luomisaikaa ja laskee onko väliaikaistiedosto vanhentunut. Ajastusfunktion koodi löytyy raportin liitteestä. (Liite 2)

#### 4.6 Sarjataulukko

Sivustolle luotu yhteisön oma sarjataulukko mahdollistaa jäsenien keskeisen kilpailun sarjataulukon sijoituksista. Jäsenien sarjataulukkotiedot tallennetaan sivuston MySQL-tietokantaan, josta ne ovat haettavissa SQL-kyselylauseilla. PHP:n avulla sivulle voidaan hakea tietoa tietokannasta. SQL-kyselyjen sekä komentojen teko PHP:llä vaatii ensiksi yhteydenluonnin tietokantaan. Yhteys, sekä tietokannan valinta suoritetaan käyttämällä seuraavia PHP:n komentoja:

```
mysql_connect("palvelimen_osoite","kayttajatunnus","salasana");
mysql_select_db("tietokannan nimi");
```

Sivuston tietokantayhteyden luomiseen vaaditut yhteystiedot on sijoitettu erilliselle PHP-tiedostolle. Tämän avulla useilla sivuilla vaadittava yhteydenluonti onnistuu lyhyellä komennolla, sekä ratkaisu parantaa sivuston tietoturvaa eristämällä yhteydenluomiseen vaadittavat tiedot käyttäjälle näkyvistä sivuista. Tietoturvaa voidaan paran-

taa sijoittamalla tietokannan yhteystiedot sisältävä PHP-tiedosto palvelimella niin, että ulkopuoliset käyttäjät eivät pääse tiedostoon käsiksi.

Sarjataulukko.php – sivu sisältää yhteisön oman sarjan sarjataulukon, linkin sarjaotte-  
luiden hallintasivulle sekä battle.net – sarjataulukon (Kuva 9). Sarjataulukot ovat laa-  
jennettuja versioita sivuston vasemmassa reunassa näkyvistä sarjataulukoista. Laa-  
jennetuissa sarjataulukoissa näytetään useampia tietokenttiä, sekä sivun käyttäjä voi  
järjestää uudelleen sarjataulukot valitsemallaan tavalla. Sarjataulukoiden sarakeot-  
sikkaa painamalla sarjataulukon jäsenet näytetään valitussa järjestyksessä, joko  
nousevasti tai laskevasti.

Etusivu

Jäsenet

Sarjataulukko

Turnaukset

Replay

Chat

#STARA\_PELLET LADDER:

Nimi	Vuorot	Häviöt	Pisteet
drLeksa	17	10	61p
Laasto	5	9	24p
kzuum	6	5	23p
imbaland	1	5	8p
Everson	2	2	8p
Kelekkis	0	0	0p
oliii	0	0	0p
TLOMek	0	0	0p

EDELLISET MATSIT:

drLeksa	Laasto	Tal'darim Altar LE
drLeksa	Laasto	Tal'darim Altar LE
drLeksa	Laasto	Tal'darim Altar LE
drLeksa	Laasto	Tal'darim Altar LE
kzuum	drLeksa	Xel'Naga Caverns

BATTLE.NET STATS:

Nimi	Liiga	Vuorot	Pisteet
Kelekkis		1	0p
kzuum		7	123p
Laasto		5	108p
imbaland		16	108p
drLeksa		0	0p

#STARA\_PELLET LADDER:

Reset	Nimi	Rotu	Matsit	Vuorot	Häviöt	Pisteet	Ratio
	drLeksa		27	17	10	61p	62.96%
	Laasto		14	5	9	24p	35.71%
	kzuum		11	6	5	23p	54.55%
	imbaland		6	1	5	8p	16.67%
	Everson		4	2	2	8p	50.00%
	Kelekkis		0	0	0	0p	0.00%
	oliii		0	0	0	0p	0.00%
	TLOMek		0	0	0	0p	0.00%

Pelaaja rankied

BATTLE.NET LADDER:

Reset	Nimi	Liiga	Rotu	Vuorot	Pisteet	Div rank
	Kelekkis			1	0p	84
	kzuum			7	123p	37
	Laasto			5	108p	30
	imbaland			16	108p	67
	drLeksa			0	0p	89

Kuva 9. Laajennetut sarjataulukot.

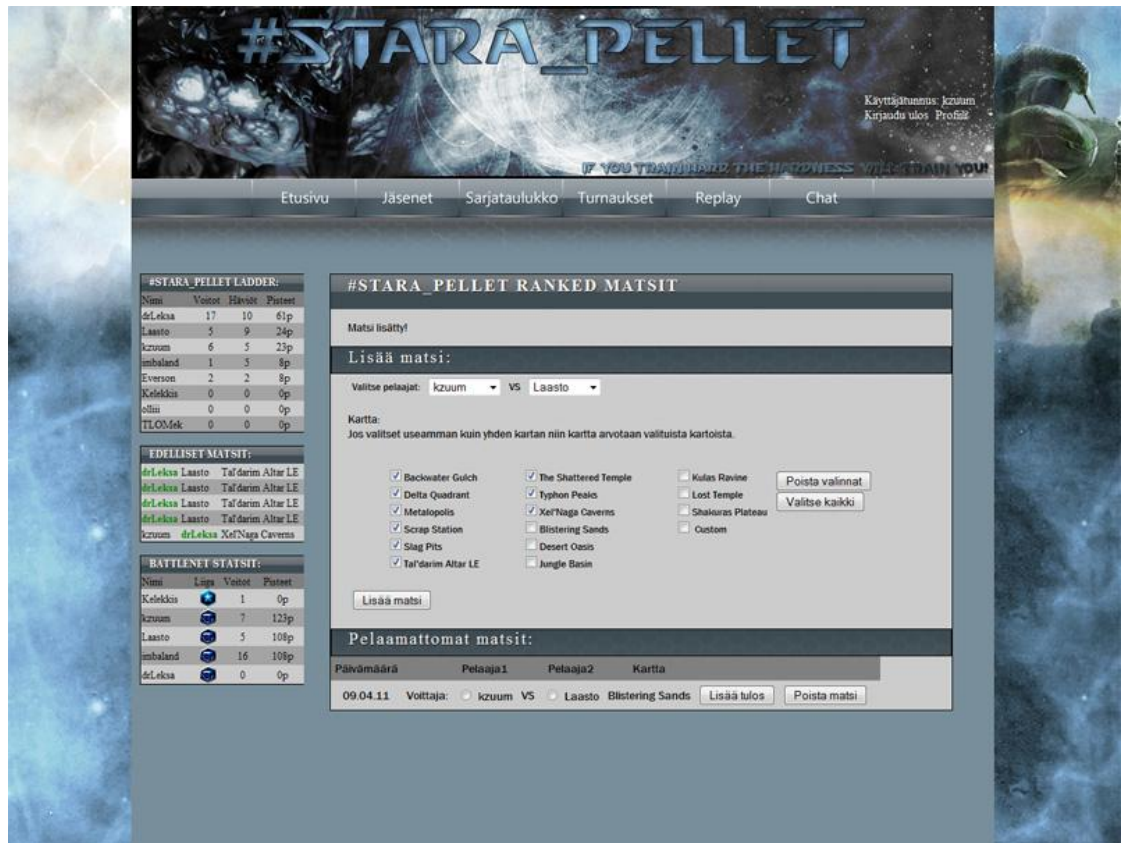
```

<table id="sarjataulukko" cellpadding="5">
    <tr id="otsikko" align="center">

<?php
// TAULUN SORTTAUS
print("<td width=\"10%\">
<a href=\"sarjataulukko.php\">Reset</a></td>");
print("<td width=\"30%\"></td>");
if(isset($_GET['suunta']) && $_GET['suunta'] == "asc"){
print("<td align=\"left\" width=\"66%\">
<a href=\"sarjataulukko.php?sortsp=nimi&suunta=desc\">Nimi
<img src=\"pics/napit/desc.png\" border=\"0\"></a></td>");}
else{
print("<td align=\"left\" width=\"66%\">
<a href=\"sarjataulukko.php?sortsp=nimi&suunta=asc\">Nimi
<img src=\"pics/napit/asc.png\" border=\"0\"></a></td>");}
...

```

Sivun PHP-koodi seuraa, onko sivulle lähetetty `$_GET['suunta']` – muuttujaa ja muokkaa sarjataulukon sarakeotsikot tuloksen mukaan. GET-metodin muuttujat näytetään sivun URL-osoitteen jatkona, jolloin käyttäjä voi esimerkiksi tallentaa sivun asetteluun selaimen kirjanmerkkeihin. Jos taulukko on järjestetty jonkin sarakkeen mukaan nousevasti, vaihdetaan sarakeotsikoihin laskevan järjestyksen valinta. Sarjataulukon järjestyksen käsittely tapahtuu muokkaamalla SQL-kyselyä sivulle lähetettyjen `$_GET` – muuttujien perusteella. SQL-kyselyjen käsittelykoodi löytyy raportin liitteenä (Liite 3).



Kuva 10. Sarjaotteluiden hallintasivu.

Sarjaotteluiden hallinta tapahtuu ranked.php – sivulla (Kuva 10), jossa sisäänkirjautuneet jäsenet voivat luoda uusia sarjaotteluita, merkata otteluiden tuloksia sekä poistaa pelaamattomaksi jääneitä otteluita. Sivulle haetaan kahteen alasvetovalikkoon yhteisön jäsenet, joista käyttäjä valitsee otteluparin. Alasvetovalikoiden alle listataan tietokantaan tallennetut kartat joista käyttäjä voi valita useamman kartan kerrallaan. Jos karttoja on valittu useampi kuin yksi, valituista kartoista arvotaan pelattava kartta.

```
if($lkm > 1){
    $kartta = array_rand(array_flip($_POST['kartta']), 1); // Jos
    karttoja on valittu useampi, arvotaan arraysta yksi arvo.
    array_flipin avulla saadaan helposti kartan nimi.
}
else{
    $kartta = reset($kartta); // Jos karttoja on valittuna yksi,
    reset antaa ensimmäisen arvon arraysta.
}
```



Hallintasivun käytettävyyteen on panostettu jotta otteluiden hallinta olisi käyttäjälle mahdollisimman vaivatonta. Käyttäjän ei tarvitse poistua sivulta sivun toimintoja käytäessään sillä tapahtumat lähetetään PHP:n POST-metodin välityksellä sivulle itselleen. POST-metodilla lähetetyt muuttujat eivät näy käyttäjälle GET-metodin tavoin sivun URL-osoitteessa. Otteluiden ylläpitotoimien koodi on raportin liitteenä (Liite 4).

#### 4.7 Replay

Yhteisön jäsenet voivat lähettää sivustolle pelitallenteen, joka listataan sivuston Replay – sivulle (Kuva 11) sivuston käyttäjien ladattavaksi. Pelitallenteita voivat ladata myös sivustolla vierailevat henkilöt.

The screenshot shows the 'Replat' page of a Starcraft 2 community website. The navigation bar at the top includes links for 'Etusivu', 'Jäsenet', 'Sarjataulukko', 'Turnaukset', 'Replay', and 'Chat'. The main content area, titled 'Replat', lists three recent matches. Each match entry consists of a table with columns for 'Pelaaja', 'Rotu', 'Väri', 'Joukkue', and 'APM'. To the right of each table, additional match details are provided, including the map name ('Kartta'), game duration ('Pelin kesto'), version ('Versio'), and a download button ('Lataa replay'). On the left side of the page, there are three smaller tables: 'STARA PELLET LADDER:', 'EDELISET MATSIT:', and 'BATTLENET STATISTIT:', each displaying player statistics and rankings.

Kuva 11. Replay – sivu.

Replay.php – sivulle haetaan palvelimella olevat Starcraft 2 – pelitallenteet. Tämä tapahtuu PHP:n glob – funktiolla. Funktiolle annetaan hakuparametrit joiden perusteella muuttujaan haetaan löydettyjen tiedostojen tiedot.

```
$directory = "./replat/";
$tiedostot = glob(" " . $directory . globistr("*.SC2Replay"),
GLOB_NOSORT);
```



Replay – sivua varten palvelimelle on ladattu GPL lisensoitu phpsc2replay – parseri (phpsc2replay, 2011). Phpsc2replay – parserin avulla sivulle saadaan haettua pelital-  
lenteiden sisältämiä tietoja kuten pelaajat, pelaajien rodut, ottelun pituus, kartta sekä  
ottelun voittaja. Phpsc2replay mahdollistaa myös monien muiden tietojen hakemisen  
pelitalienteista.

Kun pelitalienteet on haettu glob – funktion avulla, ne käsitellään yksi kerrallaan rep-  
lay parserin avulla. Pelitalienteen tiedot otetaan talteen omiin muuttujiinsa joita käyte-  
tään tulostettaessa haettuja tietoja.

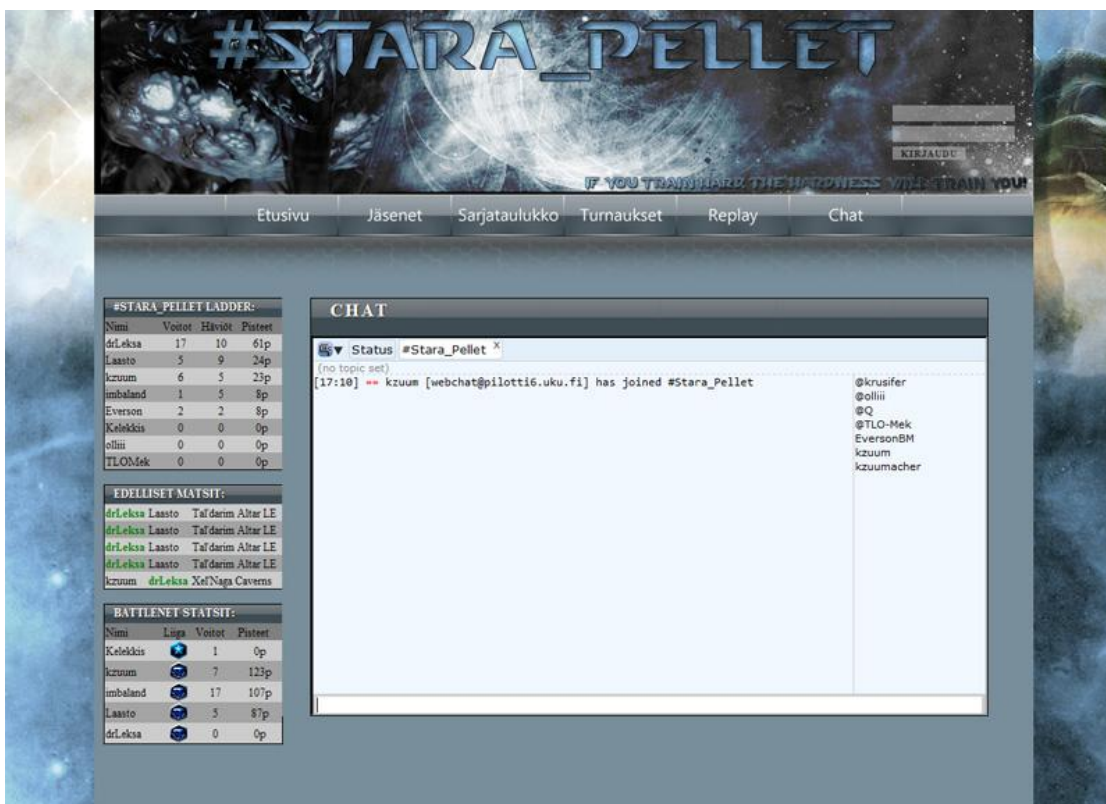
```
foreach ($tiedostot as $tiedosto) {  
    $mpqfile = new MPQFile($tiedosto);  
    $replay = $mpqfile->parseReplay();  
    $pelaajat = $replay->getPlayers();  
    $versio = $mpqfile->getVersionString();  
    ...  
}
```

Pelitalienteiden tiedonhaku ja tulostuskoodi löytyy raportin liitteenä (Liite 5).

## 4.8 Chat

Peliyhteisöllä on oma keskustelukanava IRC (Internet Relay Chat) pikaviestintäpalvelussa. Yhteisön kanava #stara\_pellet sijaitsee Quakenet – verkossa. Quakenet tarjoaa pääsyn Quakenet – verkkoon pelkällä selaimella asiakasohjelman sijaan. Tämä palvelu mahdollistaa palvelun upotuksen yhteisön sivustolle (Kuva 12).

```
<iframe align="center"
src="http://webchat.quakenet.org/?channels=Stara_Pellet"
width="716" height="400"></iframe>
```



Kuva 12. Yhteisön keskustelukanava.

## 5 SIVUSTON TESTAUS

Sivuston testaus toteutettiin suljetulla beta-testauksella. Beta-testauksen periaatteisiin kuuluu, että ohjelma tai tässä tapauksessa sivusto annetaan rajatun tai rajoittamattoman käyttäjäryhmän testikäyttöön. Suljettuun beta-testaukseen osallistuivat sivuston peliyhteisön jäsenet. Osallistuville jäsenille annettiin toimintaohjeiksi ilmoittaa sivuston kehittäjälle sivuilla tapahtuvista virheistä käytön aikana, jonka jälkeen virheet pyrittiin korjaamaan mahdollisimman nopeasti.

Sivuston kehitystyö on suoritettu erillisellä kehittämissivustolla, jonka kautta toimivat ratkaisut on siirretty yhteisön oikealle sivustolle. Näin yhteisö ei ole joutunut kärsimään kehitystyöstä johtuvista käyttökatkoksista, ja kehitteillä olevien toimintojen toimimattomuudesta. Sivuston uusista ominaisuuksista on ilmoitettu yhteisölle etusivun uutisten välityksellä. Näin he voivat tutustua ja testata uutta toimintoa, sekä antaa palautetta ja kehitysehdotuksia sivuston kehittäjälle.

## 6 POHDINTA

Opinnäytetyön tavoitteena oli saada luotua toimiva web-portaali käyttämällä mahdollisimman vähän valmiita palveluita sekä komponentteja. Itsenäisen PHP - ohjelmoinnin sekä muun web-ohjelmoinnin taitojen karttumisen kannalta tämän oli itselleni hyvä ratkaisu. Opin projektin aikana mielestäni hyvin paljon web-ohjelmoinnista ja varsinkin PHP – ohjelmoinnista.

Lähdin rakentamaan dynaamista web-sivustoa osaamatta juuri ollenkaan PHP - kieltä. Tästä johtuen sivuston tekoon kului huomattava määrä miestyötunteja. Vaikka sivustoa koodatessa tulinkin varsin usein vastaan ongelmia, mitään ylitsepääsemättömiä ne eivät olleet. Nämä vastoinikäymiset kuuluvat ohjelmointitöihin, ja olenkin iloinen että otin haasteen vastaan. Pidin sivuston tekemisestä todella paljon ja sainkin kokea hyvin usein turhautumisien jälkeen onnistumisen tunteita.

Sivuston turnausjärjestelmä on vielä kesken. Se osoittautuikin sivuston monimutkaisimmaksi ja vaikeimmaksi osuudeksi. Turnausjärjestelmään voi jo testisivustolla luoda turnauksia sekä osallistua niihin. Itse turnauksen käynnistämiseen vaadittavia toimia en ole vielä tehnyt. Turnausjärjestelmistä löytyy hyvin vähän tietoa, joten näinkin monimutkaisen systeemin rakentaminen vie aikaa.

Uskon että onnistuin luomaan toimivan web-portaalin peliyhteisölle. Yhteisön jäsenet ovat olleet tyytyväisiä sivustoon ja he tietävät että kehitän sivuja jatkuvasti. Peliyhteisöstä on ollut suuri apu sivustoa tehdessä. He ovat uskaltaneet antaa parannusehdotuksia sivuilta löytyviin toimintoihin, ja raportoineet sivuilta löytyvistä virheistä.

Pyrin saamaan sivustosta kaikin puolin toimivat turnausjärjestelmää myöten. Lisäksi tarkoituksena on kehittää profiilinhallintaa, ja tätä kautta tehdä jäsenet – sivusta kattavammat. Olenkin jopa yllättynyt kuinka innokkaana olen jatkamassa sivuston kehittämistä paremmaksi. Tavoitteena on, että kun sivusto on valmis, niin tämän sivuston pohjaa voitaisiin käyttää muiden vastaavanlaisten harrastusten sivuiksi. Opinnäytetyöksi tehty sivusto ja sen toiminnot soveltuvat aika monenlaisen harrastustoiminnan tukemiseen.

## LÄHTEET

Adobe Systems Incorporated. 2011. Mitä Photoshop tekee?. Adobe Systems Incorporated [viitattu 31.3.2011]. Saatavissa:

<http://www.adobe.com/fi/products/photoshop/photoshop/whatisphotoshop/>

Arkku.net. 2011. Ohje: Hallintapaneeli – Tietokannat. Arkku.net. Saatavissa:

<http://www.arkku.net/ohjeet&ohjeid=12583>

Blizzard Entertainment. 2011. Connect. Play. Unite. Blizzard Entertainment [viitattu 3.4.2011]. Saatavissa:

<http://eu.battle.net/en/what-is/>

Blizzard Entertainment. 2011. Starcraft II Beginner's guide. Blizzard Entertainment [viitattu 3.4.2011]. Saatavissa:

<http://eu.battle.net/sc2/en/game/guide/whats-sc2>

Bradley, A. 2011. Basic PHP Sessions. About.com [viitattu 31.3.2011]. Saatavissa:

[http://php.about.com/od/advancedphp/ss/php\\_sessions.htm](http://php.about.com/od/advancedphp/ss/php_sessions.htm)

Don HO, 2011. Saatavissa:

<http://notepad-plus-plus.org/>

Heinisuo, R. 2004. PHP ja MySQL – Tietokantapohjaiset verkkopalvelut. Helsinki: Talentum Media Oy.

Hovi, Huotari, Lahdenmäki 2003. Tietokantojen suunnittelu & ideksointi. Jyväskylä: Docendo Finland Oy.

Korpela, J. 2003. CSS-tyylit. Jyväskylä: Docendo Finland Oy.

Korpela, J. & Linjama, T. 2005. Web-suunnittelu. Jyväskylä: Docendo Finland Oy.

Open Source Initiative, 2011. The Open Source Definition [viitattu 30.3.2011].

Saatavissa: <http://www.opensource.org/docs/osd>

PhpMyAdmin devel team. 2011. PhpMyAdmin [viitattu 3.4.2011]. Saatavissa:  
[http://www.phpmyadmin.net/home\\_page/index.php](http://www.phpmyadmin.net/home_page/index.php)

Phpsc2replay. 2011. [viitattu 12.4.2011]. Saatavissa:  
<http://code.google.com/p/phpsc2replay/>

Quakenet.org. 2011. Saatavissa:  
<http://webchat.quakenet.org/>

Refactormycode.com. 2010. Cache any PHP function [viitattu 1.4.2011]. Saatavissa:  
<http://refactormycode.com/codes/77-cache-any-php-function>

SC2 Ranks. 2011. [viitattu: 3.4.2011] Saatavissa:  
<http://sc2ranks.com/api>

## LIITE 1. kirjautuminen.php

```

<?php
session_start(); //istunnon aloitus
require_once('config.php'); //tietokannan yhteystiedot

$ilmoitukset = array(); // Virhetilanteiden tallentamiseen
taulukko
$virhe = false; // tätä muutetaan jos tulee virheitä

// Yhteyden luonti
$yhteys = mysql_connect(DB_HOST, DB_USER, DB_PASS);

if(!$yhteys) {
    die('Yhteys tietokantaan ei onnistunut: ' .
mysql_error());
// Jos yhteys ei onnistu, saadaan siitä tieto
}

$db = mysql_select_db(DB_NIMI); // Valitaan tietokanta
if(!$db){
    die('Tietokantaa ei voitu valita');
}

// Funktio jolla putsataan kirjautumislomakkeen teksti.
Ehkäisee tietokannan "härkkimistä".
function siivous($str) {
    $str = @trim($str);
    if(get_magic_quotes_gpc()) {
        $str = stripslashes(@str);
    }
    return mysql_real_escape_string($str);
}

// Putsataan login-formiin syötetyt tiedot

$kayttajatunnus = siivous($_POST['kayttajatunnus']);
$salasana = siivous($_POST['salasana']);

// login-formin syötteiden tarkastus
if($kayttajatunnus == '') {
    $ilmoitukset[] = 'Käyttäjätunnus puuttuu';
    $virhe = true;
}
if($salasana == '') {
    $ilmoitukset[] = 'Salasana puuttuu';
    $virhe = true;
}
// Jos tuli virheitä lähetetään virhe-sivulle
if($virhe) {

```

```

        $_SESSION['SESS_ILMOITUS'] = $ilmoitukset;
        session_write_close();
        header("location: ilmoitus.php");
        exit();
    }

    // Tehdään kysely

    $result=mysql_query('SELECT * FROM xJasenet WHERE jNimi =
    "'.mysql_real_escape_string($_POST['kayttajatunnus']).'"
    AND jSalasana =
    "'.mysql_real_escape_string(md5($_POST['salasana'])).'"');

    // Katsotaan ovatko tiedot oikein
    if($result) {
        if(mysql_num_rows($result) == 1) {
            //Kirjautuminen onnistunut
            session_regenerate_id();
            $jasen = mysql_fetch_assoc($result);
            $_SESSION['SESS_ID'] = $jasen['jNro'];
            $_SESSION['SESS_NIMI'] = $jasen['jNimi'];
            $_SESSION['SESS_SALASANA'] =
            $jasen['jSalasana'];
            header("location: " . $_SERVER['HTTP_REFERER']);
            // palataan sivulle, jolta kirjautumiskutsu tuli
            exit();
        }else {
            //Kirjautuminen epäonnistui
            $ilmoitukset[] = 'Kirjautuminen epäonnistui: <br/>
            Tarkasta käyttäjänimesi ja salasanasasi!';
            $_SESSION['SESS_ILMOITUS']
            = $ilmoitukset;
            header("location: ilmoitus.php");
            exit();
        }
    }else {
        die("Query failed");
    }
    ?>

```



## LIITE 2. Funktion ajastus.

```

<?php
function cache_function($buildCallback, array $args = array(),
$timeoutSeconds = 3600){
// Set up the filename for the cache file
if(is_array($buildCallback)){
$cacheKey = get_class($buildCallback[0]) . '::'.
$buildCallback[1];
}else{
    $cacheKey = $buildCallback . ':' . implode(':',
$args);
}
$cacheKey .= ':' . implode(':', $args);
$file_path = "../..//phptmp/" . md5($cacheKey);

// If the file hasn't yet been created or is out of date then
call the require function and store it's result.
if(!file_exists($file_path) OR filemtime($file_path) < (time()
- $timeoutSeconds)){
    $result = call_user_func_array($buildCallback,
$args);
    file_put_contents($file_path, serialize($result),
LOCK_EX);
// Else, grab the result from the cache.
}else{
$result = unserialize(file_get_contents($file_path));
}

return $result;
}
?>

```

### LIITE 3. Sarjataulukoiden järjestyksen käsittelykoodi.

```

<?php //SARJATAULUKOIDEN SORTTAUS
if(isset($_GET['suunta'])) {
    $sortby = $_GET['suunta'];
    switch ($sortby)
    {
        case "asc":
        {
            $sqlExtra2 = "ASC";
        };break;
        case "desc":
        {
            $sqlExtra2 = "DESC";
        };break;
    }
}

// sortbnet on battle.net - sarjataulukko
if(isset($_GET['sortbnet'])) {
    $sortby = $_GET['sortbnet'];
    switch ($sortby)
    {
        case "nimi":
        {
            $sqlExtra = "ORDER BY jNimi";
        };break;
        case "liiga":
        {
            $sqlExtra = "ORDER BY jBnetLiiga";
        };break;
        case "bnetrotu":
        {
            $sqlExtra = "ORDER BY jBnetRotu";
        };break;
        case "bnetmatsit":
        {
            $sqlExtra = "ORDER BY jBnetMatsit";
        };break;
        case "bnetvoitot":
        {
            $sqlExtra = "ORDER BY jBnetVoitot";
        };break;
        case "bnethaviot":
        {
            $sqlExtra = "ORDER BY jBnetHaviot";
        };break;
        case "bnetpisteet":
        {
            $sqlExtra = "ORDER BY jBnetPisteet";
        };break;
    }
}

```

```

        case "bnetratio":
        {
            $sqlExtra = "ORDER BY jBnetRatio";
        };break;
        case "divrank":
        {
            $sqlExtra = "ORDER BY jBnetDivRank";
        };break;
    }
}

// sortsp on starapellet - sarjataulukko
if(isset($_GET['sortsp'])) {
    $sortby = $_GET['sortsp'];
    switch ($sortby)
    {
        case "nimi":
        {
            $sqlExtra = "ORDER BY jNimi";
        };break;
        case "rotu":
        {
            $sqlExtra = "ORDER BY jRotu";
        };break;
        case "matsit":
        {
            $sqlExtra = "ORDER BY jMatsit";
        };break;
        case "voitot":
        {
            $sqlExtra = "ORDER BY jVoitot";
        };break;
        case "haviot":
        {
            $sqlExtra = "ORDER BY jHaviot";
        };break;
        case "pisteet":
        {
            $sqlExtra = "ORDER BY jPisteet";
        };break;
        case "ratio":
        {
            $sqlExtra = "ORDER BY jRatio";
        };break;
    }
}
?>

```

#### LIITE 4. Sarjaotteluiden ylläpitotoimet.

```

<?php
if(isset($_POST['LisaaMatsi'])) {
if($_POST['pelaaja1'] <> $_POST['pelaaja2']) { //Tarkastetaan
onko kaksi eri pelaajaa valittuna
$yhteys = sqlConn();
$mPvm = date("d.m.y");
        $pelaaja1 = $_POST['pelaaja1'];
        $pelaaja1 =
mysql_real_escape_string(stripslashes($pelaaja1));
        $pelaaja2 = $_POST['pelaaja2'];
        $pelaaja2 =
mysql_real_escape_string(stripslashes($pelaaja2));
        $kartta = $_POST['kartta'];

        $lkm = count($kartta);
        if($lkm > 1) {
            $kartta = array_rand(array_flip($_POST['kartta']),
1); // Jos karttoja on valittu useampi, arvotaan arraysta yksi
arvo. array_flipin avulla saadaan helposti kartan nimi.
        }
        else {
            $kartta = reset($kartta); // Jos karttoja on
valittuna yksi, reset antaa ensimmäisen arvon arraysta.
        }

$sql = "INSERT INTO xMatsit VALUES (NULL,\". $pelaaja1
.\"\", \"\". $pelaaja2 .\"\", \"\". $kartta .\"\", \"\". $mPvm
.\" \")";

        mysql_query($sql);

        echo "<p>Matsi lisätty!</p>";
        mysql_close($yhteys);
    }
else { echo "<p>Et voi pelata itseäsi vastaan!</p>"; //Jos kaksi
samaa pelaajaa valittuna
}
}
?>

<?php
if(isset($_POST['LisaaTulos'])) {
$yhteys = sqlConn();
$mID = $_POST['mID'];
$voittaja = $_POST['voittaja'];
$haviaja="";
$voittajanPisteet = 3;
$haviajanPisteet = 1;
$mpelaaja1 = $_POST['mpelaaja1'];

```

```

$mpelaaja2 = $_POST['mpelaaja2'];

        if($voittaja == $mpelaaja1){
            $haviaja = $mpelaaja2;}
        else{
            $haviaja = $mpelaaja1;}

$sql = "UPDATE xMatsit SET mVoittaja='". $voittaja ."' WHERE
mID = '" . $mID . "'";
        mysql_query($sql);

$sql = "UPDATE xJasenet SET jPisteet= jPisteet + ".
$voittajanPisteet .",jVoitot= jVoitot + 1 WHERE jNimi =
'".$voittaja."'";
        mysql_query($sql);

$sql = "UPDATE xJasenet SET jPisteet= jPisteet + ".
$haviajanPisteet .",jHaviot= jHaviot + 1 WHERE jNimi =
'".$haviaja."'";
        mysql_query($sql);

        echo "<p>Tulos kirjattu!</p><br/>";
        echo "<p>Pisteet jaettu!</p><br/><br/>";
        mysql_close($yhteys);
    }
?>

<?php
if(isset($_POST['PoistaRivi'])) {
    $yhteys = sqlConn();
    $mID = $_POST['mID'];

    $sql = "DELETE FROM xMatsit WHERE mID = '". $mID
. "'";
    mysql_query($sql);

    echo "<p>Matsi poistettu!</p><br/>";
    mysql_close($yhteys);
}
?>

```

## LIITE 5. Pelitallenteiden tiedonhaku ja tulostus.

```

<?php
$directory = "./replat/";
$tiedostot = glob(" " . $directory . globistr("*.SC2Replay"),
GLOB_NOSORT);
array_multisort(array_map('filemtime', $tiedostot),
SORT_NUMERIC, SORT_DESC, $tiedostot);
$jako;
$r = 1;
foreach ($tiedostot as $tiedosto) {
    $mpqfile = new MPQFile($tiedosto);
    $replay = $mpqfile->parseReplay();
    $pelaajat = $replay->getPlayers();
    $versio = $mpqfile->getVersionString();
    $versio = substr_replace($versio, "", 6); // Otetaan
    ylimääräiset versionumeroinnit pois
    $jako;
    $i = 1;

    print("<table border=\"0\"><tr>");
    $obsString = "";
    $obsCount = 0;

    echo "<td><table border=\"0\" id=\"replat\"
    cellpadding=\"5\" style=\"margin-left: 20px; margin-
    right: 10px; margin-top: 20px;\"><tr
    id=\"otsikko\"><th>Pelaaja</th><th>Rotu</th><th>Väri<
    /th><th>Joukkue</th><th>APM</th></tr>\n";

    foreach($pelaajat as $value) {
        if ($value['isObs']) {
            if ($obsString == "")

            $obsString = $value['name'];

        else

        $obsString .= ', ' . $value['name'];

        $obsCount++;

        continue;
    }
    if ($replay->isWinnerKnown())
    $wincolor = (isset($value['won']) && $value['won'] ==
    1)?0x00FF00:0xFF0000;
    else
    $wincolor = 0xFFFFFFFF;

```

```

if ($value['isComp'] && $replay->getTeamSize() !=
null)
$difficultyString = sprintf(
("%s"),SC2Replay::$difficultyLevels[$value['difficulty
']]);
else

$difficultyString = "";

if ($i % 2){
$jako = "<tr id=\"pariton\">";}
else {
$jako = "<tr id=\"parillinen\">";}

if($value['won'] == "1"){

$voittajat[$r] .= " " . $value['name'];}

echo sprintf("%s<td>%s</td><td>%s</td><td><font
color=\"#%s\">%s</font></td><td>%s</td><td>
style=\"text-align: center\">%d</td></tr>",

$jako,
$value['name'].$difficultyString,
$value['race'],
$value['color'],
$value['sColor'],
($value['team'] > 0)?"Team ".$value['team']:"- ",
($value['team'] > 0)?(round($value['apmtotal'] /
($replay->getGameLength() / 60)):0);

$i++;
}

echo "<tr><td colspan=\"5\" align=\"center\"><a href=\"".
$tiedosto . "\">Lataa replay</a></td></tr></table></td>";

echo sprintf("<td valign=\"top\"><br/><font
id=\"replayinfo\"><strong>Kartta:</strong> %s<br/>
<strong>Pelin kesto:</strong> %s<br/>", $replay-
>getMapName(), $replay->getFormattedGameLength());

echo sprintf("<strong>Versio:</strong> %s<br/>
<strong>Pelimuoto:</strong> %s<br/>", $versio, $replay-
>getTeamSize());

echo sprintf("<strong>Pelattu:</strong>
%s<br/></font>", date('d.m.Y \k\l\o H:i' , $replay->getCtime()));

```

```
print("<div><div style=\"display: inline-block; margin: 10px
0px 10px 0px;\"><input type=\"button\" id=\"btn\".$r.\"\"
value=\"Näytä voittaja\" onClick=\"voittaja('voittajat\".$r.\"',
'btn\".$r.\"')\"></div>");
```

```
print("<div id=\"voittajat\".$r.\"\" style=\"visibility:hidden;
margin-left: -105px; display: inline-block;\"><font
id=\"replayinfo\"><strong>Voittaja:</strong>
\".$voittajat[$r].\"</font></div></div></td><td
width=\"*\"></td></tr>");
```

```
$r++;
print("</table><br/><hr/>");
}
?>
```

```
<p><strong>Lähetä replay:</strong></p>
<form id="margin20" action="upload.php" method="post"
enctype="multipart/form-data">
<label for="file">Replay:</label>
<input type="file" name="file" id="file" />
<br />
<input type="submit" name="submit" value="Lähetä" />
</form>
<br/>
<br/>
</div>
```

```
</div>
<br style="clear:both" /> <!-- Että vasemman puolen taustaväri
jatkaa alas asti! -->
</div>
```

```
<script type="text/javascript">
function voittaja(teksti, nappi)
{
    var teksti = document.getElementById(teksti);
    var nappi = document.getElementById(nappi);
    teksti.style.visibility = 'visible';
    nappi.style.visibility = 'hidden';
}
</script>
```